

Tasca_M2_T01

April 11, 2023

0.0.1 Sprint 2

Tasca M2 T01

0.0.2 Exercici 1.

Crea una llista que agrupi els mesos de l'any en trimestres (1T: gener, febrer i març; 2T: abril, maig, juny...), és a dir, una llista amb 4 llistes dins.

```
[1]: # Dues maneres de crear llistes entre 'brackets' i utilitzant el constructor.  
# Al final comprovem que són del tipus llista i mostrem la llista de llistes_  
↳exigida a l'exercici.
```

```
T1 = ['Gener', 'Febrer', 'Març']  
T2 = list(('Abril', 'Maig', 'Juny'))  
T3 = ['Juliol', 'Agost', 'Setembre']  
T4 = list(('Octubre', 'Novembre', 'Dezembre'))  
  
mesos = [T1,T2,T3,T4]  
  
for x in mesos:  
    print (type(x))  
  
print()  
print('La llista mesos consisteix en una llista (mesos de l\'any) de llistes_  
↳(dividida en trimestres): ')  
print (mesos)
```

```
<class 'list'>  
<class 'list'>  
<class 'list'>  
<class 'list'>
```

La llista mesos consisteix en una llista (mesos de l'any) de llistes (dividida en trimestres):

```
[['Gener', 'Febrer', 'Març'], ['Abril', 'Maig', 'Juny'], ['Juliol', 'Agost',  
'Setembre'], ['Octubre', 'Novembre', 'Dezembre']]
```

```
[2]: # Una altra forma de crear la llista de llistes és utilitzant el mètode append
      ↪(també extend())
      mesos2 = []
      for i in mesos:
          mesos2.append(i)

      mesos2
```

```
[2]: [['Gener', 'Febrer', 'Març'],
      ['Abril', 'Maig', 'Juny'],
      ['Juliol', 'Agost', 'Setembre'],
      ['Octubre', 'Novembre', 'Dezembre']]
```

```
[3]: # I comprovem que és mesos2 és del tipus llista i que és igual que mesos.

      print (type(mesos2)==list)
      print(mesos2 == mesos)
```

True

True

0.0.3 Exercici 2

Crea un codi que et permeti accedir a:

- El segon mes del primer trimestre.
- Els mesos del primer trimestre.
- Setembre i octubre.

```
[4]: import numpy as np
```

```
[5]: # Amb el shape de numpy podem comprobar que la nostra llista mesos és de (4,3)
      ↪-> 4 trimestres i 3 mesos
      np.shape(mesos)
```

```
[5]: (4, 3)
```

```
[6]: # Per accedir al segon mes (posició 1) del primer trimestre (posició 0):
      mesos[0][1]
```

```
[6]: 'Febrer'
```

```
[7]: # Per accedir als mesos del primer trimestre només utilitzarem la primera
      ↪dimensió:
      mesos[0]
```

```
[7]: ['Gener', 'Febrer', 'Març']
```

```
[8]: # Per accedir a Setembre i Octubre:  
print(mesos[2][2], mesos[3][0])
```

Setembre Octubre

0.0.4 Exercici 3

Crea una llista amb nombres desordenats i respon a les següents preguntes:

- Quants números hi ha?
- Quantes vegades apareix el número 3.
- Quantes vegades apareixen els nombres 3 i 4?
- Quin és el número més gran?
- Quins són els 3 números més petits?
- Quin és el rang d'aquesta llista?

```
[9]: import random
```

```
[10]: # A partir d'una llista buida, amb el mètode append generarem una llista de 50  
      ↪ nombres enters entre 0 i 10  
  
integers=[]  
for i in range(50):  
    res = random.randint(0,10)  
    integers.append(res)  
  
print(integers)
```

[5, 5, 3, 2, 9, 1, 10, 0, 7, 9, 8, 8, 8, 10, 8, 5, 8, 1, 7, 5, 5, 10, 6, 5, 3,
10, 5, 10, 0, 6, 10, 5, 7, 5, 5, 0, 0, 6, 4, 10, 3, 0, 6, 3, 5, 10, 0, 3, 4, 5]

```
[11]: # Una altra forma més directe utilitzant el randomitzador de numpy i  
      ↪ transformant-la en llista  
  
integers2 = np.random.randint(0,10,50)  
integers2 = list(integers2)
```

```
[12]: # Quants números hi ha?  
  
print(len(integers))
```

50

```
[13]: # Quantes vegades es repeteix el tres?  
  
tresos = integers.count(3)  
tresos
```

[13]: 5

```
[14]: # Quantes vegades apareixen els nombres 3 i 4?
```

```
tres_quatre = integers.count(3) + integers.count(4)
tres_quatre
```

```
[14]: 7
```

```
[15]: # Quin és el número més gran?
```

```
gran = max(integers)
gran
```

```
[15]: 10
```

```
[16]: # Quins son els 3 números més petits?
```

```
# Eliminarem els duplicats, ordenarem la llista de menor a major i mostrarem
↳els tres primers elements:
```

```
unique_int = set(integers) # Aquest mètode transforma la llista en set
unique_int = list(unique_int) # Tornem a transformar en llista ja ordenada
unique_int[0:3] # Mostrem els tres més petits
```

```
[16]: [0, 1, 2]
```

```
[17]: # Quin és el rang d'aquesta llista?
```

```
rang = max(integers) - min(integers)
rang
```

```
[17]: 10
```

0.0.5 Exercici 4

Crea un diccionari de la següent forma i respon a les preguntes:

- Afegeix alguna fruita més
- Quant han costat les peres en total?
- Quantes fruites hem comprat en total?
- Quina és la fruita més cara?

```
[18]: compra = { "Pomes" : {"Qty": 5, "€": 0.42}, "Peres" : {"Qty": 3, "€": 0.66} }
```

```
[19]: # Afegeix alguna fruita més
```

```
compra['Pinya'] = {'Qty':5, '€':0.98}
compra
```

```
[19]: {'Pomes': {'Qty': 5, '€': 0.42},
      'Peres': {'Qty': 3, '€': 0.66},
      'Pinya': {'Qty': 5, '€': 0.98}}
```

```
[20]: # Quant han costat les peres en total?

preu_peres = compra['Peres']['Qty'] * compra['Peres']['€']
preu_peres
```

```
[20]: 1.98
```

```
[21]: # Quantes fruites hem comprat en total.

peces = []
for fruites, atributs in compra.items():
    qt = atributs['Qty']
    peces.append(qt)

print('Hem comprat un total de {} tipus de fruita'.format(len(compra.keys())))
print('')
print('Hem comprat un total de {} peces de fruita'.format(sum(peces)))
```

Hem comprat un total de 3 tipus de fruita

Hem comprat un total de 13 peces de fruita

```
[22]: import pandas as pd
```

```
[23]: # Quina és la fruita més cara?
# Per facilitar la operació (?) creem un df amb tres columnes, fruita, preu_
↳ individual i preu total.
# El df està buit i anem incorporant les dades utilitzant un dels mètodes_
↳ iterables per accedir al diccionari.

df = pd.DataFrame(columns=['fruita', 'preu_individual', 'preu_total'])
for fruites, atributs in compra.items():
    pr = (atributs['€'])
    qt = atributs['Qty']
    # print(fruitas, pr, qt*pr)
    df.loc[len(df)] = [fruites, pr, qt*pr]
```

```
[24]: # Observem que, per preu individual la fruita més cara és la pinya, després la_
↳ pera i finalment la poma.

df.sort_values(by=['preu_individual'], ascending=False)
```

```
[24]:   fruita  preu_individual  preu_total
      2  Pinya              0.98        4.90
      1  Peres              0.66        1.98
      0  Pomes              0.42        2.10
```

```
[25]: # Ordenant pel preu total, la pinya segueix essent la més cara.
      #En aquest cas el segon lloc és per la poma i finalment la pera

      df.sort_values(by=['preu_total'], ascending=False)
```

```
[25]:   fruita  preu_individual  preu_total
      2  Pinya              0.98        4.90
      0  Pomes              0.42        2.10
      1  Peres              0.66        1.98
```